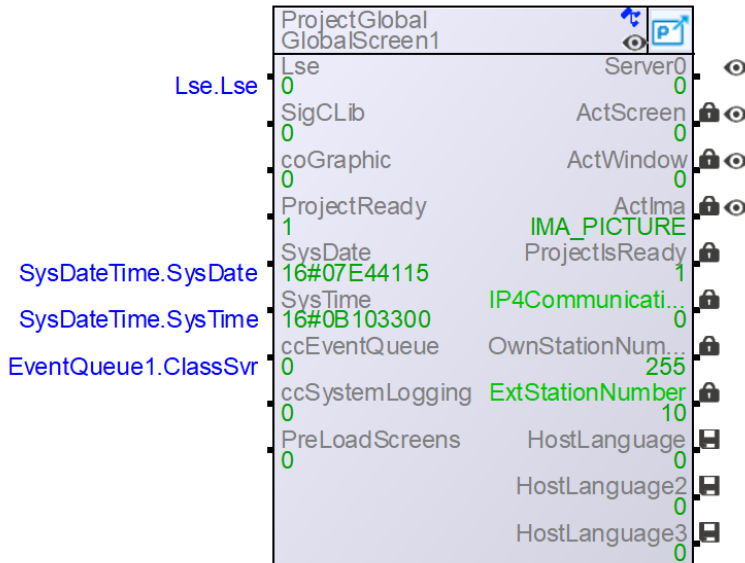


# ProjectGlobal



This class handles global project information and settings.

**ATTENTION:** needs \_LSE version 0.114 or higher.

## Interfaces

### Servers

<b>Server0</b>	Inherited from the class _Global
<b>ActScreen</b>	Shows the number of the active screen
<b>ActWindow</b>	Shows the number of the active window
<b>ActIma</b>	Shows, whether a screen or a window is active
<b>ProjectIsReady</b>	Output: 1 = Project is Ready (LSE Ready)
<b>IP4Communication</b>	Command channel to get the IP address from the station number
<b>OwnStationNumber</b>	LSE-Station Number of HMI. If initvalue=-1, value is determined automatically.

	<p>Automatic mode: Data of IPC.INI is browsed and station numbers are extracted. If more than one internal station is defined, ProjectGlobal will use the station with the highest value.</p> <p>IMPORTANT: In automatic mode, Station number is not valid before call of IF_ProjectReady!</p> <p>Automatic mode does not support substations (see t_ipc.h, #define UC_MULTICPU)</p>
<b>ExtStationNumber</b>	<p>Station number of the process station.</p> <p>Also see: OwnStationNumber.</p>
<b>HostLanguage</b> <b>HostLanguage2</b> <b>HostLanguage3</b>	<p>System wide host language</p> <p>Is set after each Init and with each writing access to this server.</p> <p>0 ... Disable this functionality x ... Windows ID of the language or index in the LSE language list (0-based)</p> <p>Example for Windows ID: 1033 = English (United States) 2057 = English (Britain) 1031 = German (Standard Germany) 3079 = German (Austria) 1043 = Dutch (Standard)</p> <p>Attention: the language must exist in the LSE project!</p>

## Clients

<b>Lse</b>	Inherited from the class _Global
<b>SigCLib</b>	Object channel to the operating system interface SigCLib (created automatically)
<b>coGraphic</b>	Object channel Graphic (created automatically)
<b>ProjectReady</b>	TRUE = visualization booted successfully
<b>SysDate</b>	Connection to read the current date
<b>SysTime</b>	Connection to read the current time
<b>ccEventQueue</b>	Command channel to an object of the class EventQueue (optional)
<b>ccSystemLogging</b>	Command channel to an object of the class SystemLogging (optional)
<b>PreLoadScreens</b>	Client to the PreLoadingScreens table class (optional). By connecting this client, the PreLoadingScreens functionality is activated.

## Global Methods

<b>IF_WindowStart</b>	<p>Inherited from the class _Global</p> <p>Is called on opening/closing windows and switching screens. Is used in this class to determine the current screen number and the number of the start screen.</p> <p>IN: whoami ... picture, window or menu and number  IN: pic ... pointer to picture, window or menu  IN firsttime ... is set to TRUE on the first call</p>
<b>IF_WindowRun</b>	<p>Inherited from the class _Global</p> <p>This method is called cyclically for each picture, window or menu. Is used in this class to save the current date and time.</p> <p>IN: whoami ... picture, window or menu and number  IN: pic ... pointer to picture, window or menu  IN: event ... pointer to possible events, keyboard or touch activities</p>
<b>IF_ProjectReady</b>	<p>Inherited from the class _Global</p> <p>This method is called after loading the project. Graphics are already initialized but not output to the screen. Is used in this class to enter the events for switching on and off the control in the event journal and to get the IP address of the station via the station number.</p> <p>IN: state ... TRUE: project loaded successfully  FALSE: project could not be loaded</p> <p>OUT: retcode ... TRUE: application should be started  FALSE: application should be stopped</p>
<b>IF_InputIsSaved</b>	<p>Inherited from the class _Global</p> <p>This method is called, when an input was accepted successfully. Is used in this class to react on different inputs and make entries in the event journal.</p> <p>IN: pi ... pointer to the input information  IN: pr ... pointer to the variable information</p>
<b>IF_Write</b>	<p>Inherited from the class _Global</p> <p>The method is called before a new value is written to the system (input is finished). So the user can avoid, that a value is written. Is used in this class to check, whether the value change should be entered in the logbook.</p> <p>IN: pres ... pointer to the input data  IN: pvar ... pointer to the variable information</p> <p>OUT: retcode ... TRUE: value may be written  FALSE: value must not be written</p>

<b>GetStartPicture</b>	<p>Returns the screen number of the start screen.</p> <p>OUT: uiStartPictureNr ... number of the start screen</p>
<b>GetIpAdressOfStation</b>	<p>Gets the IP address with the help of the station number and returns a pointer to a text with the IP address.</p> <p>IN: plpAdress.. pointer to a text with the IP address of the station</p> <p>OUT: retcode ... 0: project not completely loaded -1: no IP address 1: OK</p>